Algorithmic Strategies of Routing Seamless Parcel Delivery

*Note: A Simulation Optimisation Study to Reduce Parcel Delay and Left-over Parcels

Debmalya Deb
Msc in Data Analytics (School of Computing)
National College of Ireland
Dublin, Ireland
x21242101

***Abstract*—This study proposes a business model for the "We-Doo" start-up and optimises multiple simulation approaches over the delivery services in a commuter township that expanded all over the county with an electronic scooter ranging between 35km to 45km per day whose standard working time is 3 hours per day(18:00-21:00). This paper sets its map(M), a warehouse centre and selects 50 nodes and 100 customers. Further, the delivery per customer per day(p) has been bounded in the range of 0.2 to 0.3. A*algorithm and Floyd-Warshall algorithm have been explored to find the shortest distance. By generating the round trip and with the help of a recorder this study creates its model flow from delivery centre to distribution. Many large simulation models are being triggered out of those 3 best models are being proposed as business models by setting up the range of 10, over 50 days for 100 customers and analysing daily working time, daily leftover parcels and delays in parcel delivery.**

***Index Terms*—Simulation, Optimisation, P value, Daily Working Time, Parcel Delay, Shortest Path, Event Graph**

I. Introduction

The growth of e-commerce platforms and their logistic models are nowadays exponentially growing with an increasing amount of people moving towards online shopping. People expect their parcel's delivery within the next day or the day after. Not only that the logistics companies are trying to get over the legacy delivery system and embrace the algorithmic simulation models just to commit the customer to the exact timeline of their ordered parcels to maintain the company's reputation and goodwill.

This study is dedicated to building a business model for Routing Seamless Parcel Delivery for a start-up called "We-Doo". This parcel delivery service works in a township basis model that grows up all over the country.

"We-Doo" basically generates a Map with 50 nodes and 100 customers which is considered a township and finds the shortest distance out of it with the A*algorithm and Floyd-Warshall algorithm. After that, it checks the possible round trips to deliver the on-time parcels, leftover parcels or delayed parcels out of these three parameters the last two are the prime ones of this project and with the help of a recorder it memories the previous shortest path and keeps on searching for the following shortest distance.

Successfully generating and tuning all constraints this study runs multiple simulations over the 500 days range, setting the parameter of per customer parcel(p) 0.2 to 0.3 with the help of a single electronic vehicle(scooter) which covers 30km, 35km, 40km and 45km respectively per day and runs the simulation out of it. One thing to note in this regard is that the delivery service commences only 3 hours per day(18:00-21:00) and during simulation it has been taken care of whether the bikes are running within that range or not and make sure that the average mean working time must co-inside within this 180 minutes range per day.

In the delivery service, the most challenging part is to diminish the daily delayed parcel and daily leftover parcels. Many times it does not depend on the product in-transit process, many times it has been seen that the customer is not at home or they are not responding to the call due to some reasons. Due to miss attempt at the delivery, the delivery agent has to come back again to deliver the same parcel. But for that, the delivery agent sometimes lost his nearest or shortest track for other delivery and sometimes the agent needs to extend the working time to cover the longest distance or he might need a longer working bike to cover such a long distance. But the company has to deliver the parcels with keeping in mind that the leftover parcels would not be high and delays in parcel delivery numbers would be less in percentage with respect to days.

Overall, this study finds 3 best-fit simulation models for the "We-Doo" which can be deployed in the township. Out of the simulated models it has been observed that the 30km bike range and p=0.2 models are the most optimised models based on the generated map. In that simulations, the soft constrain working time some days are exceeding the 180 minutes binding and altogether the average mean is within that particular region. Not only that the maximum optimisation of leftover parcels and delay in delivery are being achieved in those models.

II. Literature Review

In this section, this study refers to some peer-reviewed published research papers on modelling simulation and optimisation on routing delivery parcels.

Stochastic and Dynamic features approaches have been addressed by moving the freights to an urban sector to improve in-parcel delivery and to reduce the problem of the crowded delivery agent at the warehouse by using multiple delivery options. The essential benefits of using Stochastic and Dynamic features approaches are less travel time and cost reduction. With three-fold strategies, the authors are trying to address this issue. Firstly, they are creating a multi-staged stochastic model and accumulate the elements which cause problems during the delivery and they call it a Dynamic and Stochastic Vehicle Routing Problem with Time Windows(DS-VRPTW). Secondly, they have created a simulation model to address this DS-VRPTW problem in real time with different realistic scenarios. Metaheuristic approaches are being explored regarding this. And lastly, they have incorporated large modules by involving heady duty vehicles on this ground for example vans, cargo bikes etc [1].

This study finds the above paper very practical because before generating their own model they are accumulating the problems gathered by various surveys during real-time delivery, more precisely speaking when the map is huge. One similarity has been found between these two papers is of using Meta heuristic algorithm to find the shortest distance

The next paper which could connect this problem statement would be a ”Travelling Salesman Problem(TSP)”. The author of this paper [2] targets to cut down the transportation cost without losing on-time delivery criteria and customer satisfaction. Heuristic and meta-heuristic algorithms are being used to determine the shortest distance path which also helps to reduce the computation time during optimisation. In their case study, they build a closed loop-based route that is where the starting point and ending point will be the same so that the round trip mechanism can be implemented. They build the shortest loop by connecting short-distance nodes in terms of delivery points using the shortest Hamiltonian cycle. Not only that the authors limit the vehicle's daily working kilometers. This business model believes in skipping some of the nodes by some means of penalties just to reach on time and on track to the rest of the nodes.

The above paper influenced this study with respect to finding the shortest path and making the loop. One impactful parameter has been noticed which is skipping the nodes with some penalties just the reach to achieve the broader goal. In this study also the author could refer to it as its future work, just like to diminish the delay time the delivery agent could skip some of the nodes by paying some penalties.

III. Methodology

This optimisation and simulation paper uses Python code from generating a map to a few large simulations to build a model and now the author is going to go through its step-by-step methodology in the following manner,

A. Map Generation

In the first stage of map generation, firstly a city of 10km diameter has been selected and 50 nearest node points have been randomly spotted keeping in mind that no new points are nearer existing points. Giving no order preference to any of the node's lines, pair of nodes are getting connected with the line. A triangulation algorithm [3] has been employed on this set of nodes just to form a grip or mesh which helps to create small loops. The triangulation algorithm takes 3 random points to create its first phase and to create a mesh or loop in the second phase it takes any of the auxiliary points as a starting point from phase one and creates another triangular mesh. In this way, the map connects all 50 nodes in an algorithmic manner. Now it's time to randomly choose a point/node as a warehouse or delivery centre and with that 100 points have been chosen as the delivery point or customer location around the delivery centre. A ”generateData” function has been called with 50 nodes, and 100 customers and as a seed value this paper passed the author's last 4 digits of the student id to create a unique map. And seed value 0 has been set for 11 customers and 6 nodes just to create a ”sampleData” map for further testing down the line. This function has 3 major parameters which are M(V, E), W and C. Here M refers to Map. V and E refer to the vertices and edges of the map. W and C are the warehouse location and customer location respectively.

Finally, the ”sampleData” map and the Actual map for the final simulation have been created in Fig 1, and Fig 2 respectively. These maps are loaded and saved as sampleData.pickled, data.pickled respectively for further simulation.

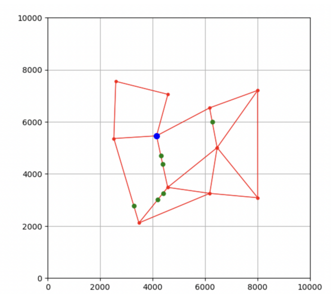

Fig. 1. Testing map with 11 customers and 6 nodes

B. Generate Parcels

Now this project introduces a parameter called p(average number of parcels per customer) ranges between 0.2 to 0.3. A ”generateParcels” function has been set with days, no of customers and p-value. While testing this function over the ”sampleData” map by setting 500 days, 100 customers and p=0.2 this study finds an average of 100.83 parcels per customer and 20.17 parcels per day(Fig:3).

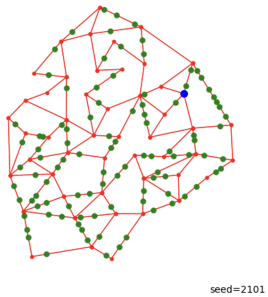

Fig. 2. Actual Map for 50 nodes and 100 customers

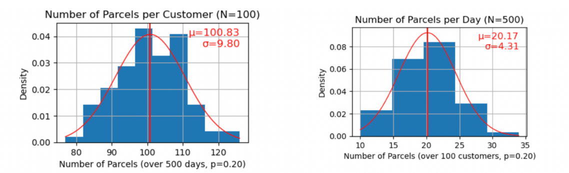

Fig. 3. Parcel Generation

C. Finding Shortest Path

A^* algorithm and Floyd-warshall(FW) algorithm have been used to generate and check the shortest path. A^* algorithm[1] finds the cheapest and least distance path from the starting node to the termination point. It follows

$$f(n) = g(n) + h(n) \tag{1}$$

function to reach its goal. Where g(n) is the cost of the path from the starting node to n^th node and h(n) is the heuristic function to find the cheapest path from n.

And the dynamic Floyd-warshall[2] algorithm compares all possible vertices from a start node and selects only the shortest path out of it.

Fig:4 clearly shows for this map both of the algorithms find 2439m as its shortest path. So as the FW algorithm takes less computation time than A* so that in this study we would proceed with FW methods

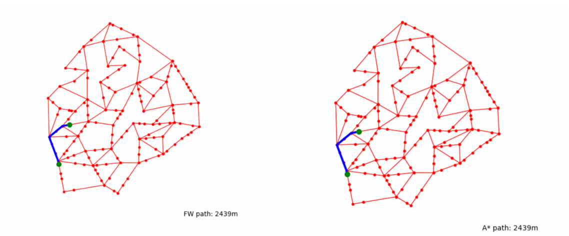

Fig. 4. Finding Shortest Distance

[1]https://en.wikipedia.org/wiki/A*$_{s}earch_{a}lgorithm$
[2]https://en.wikipedia.org/wiki/Floyd–Warshall$_{a}lgorithm$

D. Modelling via Event Graph

A delivery system identifies multiple entities just as Parcels (moving), Customers (static), Driver (Unique Resource) and Delivery Centre (Unique Resource) starting from the order placed to delivery. It also includes some variables and parameters for example destination, location, delivery centre, maxtourlenth etc and status, at home, parcel receive, left over, address etc respectively.

All these below-mentioned event graphs are built-in Miro[3]

1) Event graph for Parcel: Fig 5[4] represents the event graph for a parcel where the parameter and variable are destinations(Example: delivery centre, delivery location) and the status(order placed, in transit, handed over to delivery agent etc)

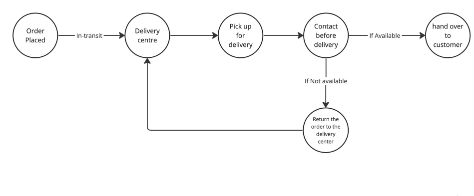

Fig. 5. Event graph for Class Parcel

2) Event graph for Customer: The event graph for customers has been shown in Fig 6[5]. The parameters and variables are location(position of the parcel) and at home(if the customer is at home), answers door(response to delivery agent's call), and parcel received(package delivered) respectively.

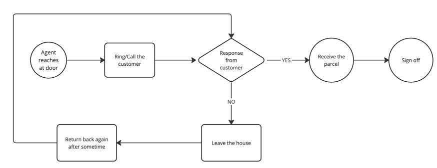

Fig. 6. Event graph for Customer

3) Event graph for Driver: In Fig:7[6] event graph for the driver is referred and as variables parcelForDelivery(parcel out for delivery), tour and customerLocation(customer is at home or not) have been taken.

4) Event graph for Delivery centre: The delivery centre's event graph has been shown in Fig:8[7]. DeliverCentre(accept parcel) and maxTourLength(total distance of travel) have been taken as parameters. And leftOver(those parcels are not being delivered), parForDelivery, addresses(customer's location), and tour are being taken as variables.

[3]https://miro.com/app/dashboard/
[4]https://shorturl.at/dgUY6
[5]https://shorturl.at/eixH4
[6]https://shorturl.at/ervO8
[7]https://shorturl.at/aBIN5

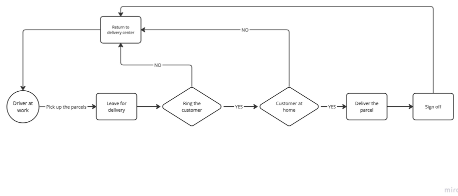

Fig. 7. Event graph for Driver

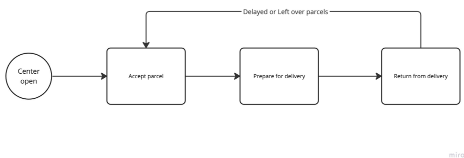

Fig. 8. Event graph for Delivery centre

E. Modelling Simulation

1) Simulation Parameters: One of the co-founders of the "We-Doo" start-up shared his experience and based on that the parameters are being set.

- AVERAGE_SPEED- The electronic bike which is used in delivering the parcels runs at 15km per hour.
- AVERAGE_TIME_ANSWER_DOOR- The average mean time of handing over the parcels from the delivery agent to the customer is 40 sec which is calculated in expovariate distribution.
- AVERAGE_TIME_HANDOVER- And another exponential distribution average mean time of 10 seconds is being taken as additional handover time.
- AVERAGE_TIME_SIGNOFF- The average sign-off time has been taken as 10 sec.
- PREP_TIME_PER_PARCEL- The preparation time(routing plan, parcel sorting, at the delivery centre and packing the cargo) has been presumed to be 50 sec per parcel.
- END_OF_DAY_TIME- End procedure which includes the time for bike charging, reporting to the delivery centre and cargo set up has been assumed to be taken 10 minutes per parcel.
- RETURN_TIME_PER_PARCEL- The undelivered parcel sorting time has taken 30 sec per parcel to be delivered.

2) Simulation Function: A Simulation function "simulation" has been created by passing map geometry(M, W, C), simulation days, parcels per customer(p), bike limit, probably that the customer is not at home(set as 0.1), max wait time at the customer's door. This model function has been verified using sampleData map by generating all customer location points etc starting from day 0 to the last day of the delivery.

Another two functions, multisimulation and loadSimulations are being created using the same arguments mentioned above. These are for getting large simulations by setting up a range of 10 and loading and storing the large simulation respectively in the rec folder after each trial.

F. Large Simulation Analysis

This study runs multiple larger simulations by using the map stored in data.pickled fild and by testing and verifying all the parameters the author finds 3 best-fit models for final comparison and one more extra model to show the out-range simulation to justify the best model's significance. Statistical analysis has been performed on total working time, daily leftover parcels, and delayed delivery of parcels.

The simulation models parameters are as follows,

- **p-value-** 0.2, 0.21, 0.22
- **Limit-** 30km, 35km
- **Total no of Days-** 500(by setting the range 10 in multi simulation and large simulation function)

1) Simulation 1(p-0.2, limit-30000, days-50, range-10): By setting up the above parameters, in each simulation, the total number of parcels has been generated ranging from 976 to 1052 over 50 days to 100 customers. And likewise, 10 simulations of seed values 0 to 9 have been performed.

Fig 9 shows the daily working time of the delivery agent for the first simulation which is on average 175.82 minutes per day. And it is under the daily working limit. And also the line chart is not following any trend.

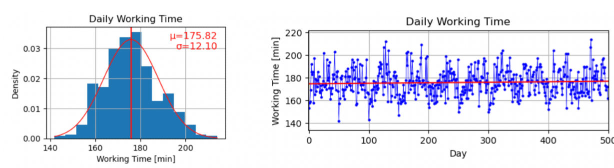

Fig. 9. Daily Working Time

The daily tour length time as shown in Fig 10 is 29774.17 meters which supports the 30000m bike that has been utilised at its optimum point and the line plot also shows a continuous one.

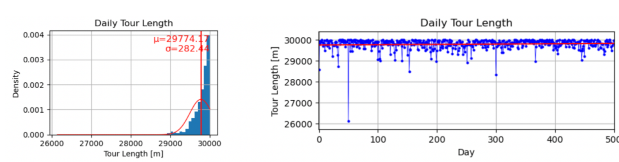

Fig. 10. Daily tour length

Fig 11 captures the most crucial parameter, daily leftover parcels. The daily left-over parcel's mean is 37.26 and the line plot is not steep which means it does not show any uptrend.

As per Fig 12, it is clearly visible that the daily parcel delay is high on day 0 and it has increased on day 1, after day 2 it has started decreasing. On day 11 it reached 0.1% and from day 14 it is 0%

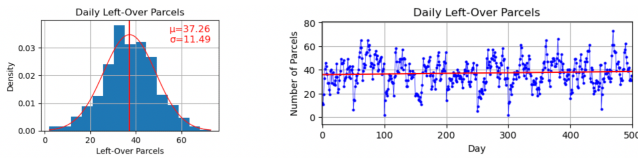

Fig. 11. Daily leftover parcels

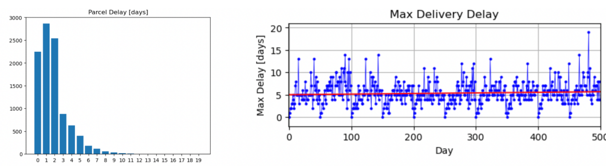

Fig. 12. Max delivery delay

2) Simulation 2(p-0.21, limit-30000, days-50, range-10): With the above parameters, the model generates parcels ranging from 1014 to 1080 over 50 days to 100 customers for a single seed value and range. Likewise, 10 range simulation has been carried out for the statistical analysis.

Fig 13 shows the histogram and line plot of the daily working time for simulation 2. The average mean time shown here is 178.07 minutes which has slightly increased to simulation 1. Significantly no trend has been observed throughout the simulation.

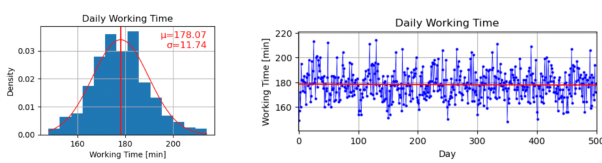

Fig. 13. Sim 2 Daily Working Time

Here as well 30km range bike has been chosen and from the histogram and line plot it is straightforward that this simulation is statistically significant and optimised because the bike runs at its ultimate range throughout the delivery days, Fig 14.

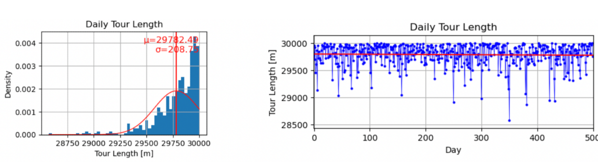

Fig. 14. Sim 2 Daily tour length

In Simulation 2, the mean of the daily left over parcels has slightly increased as compared to Simulation 1 as per Fig 15. But a noticeable factor has been observed which is a slight downtrend which could make this model the best one at the end

A little bit longer parcel delay percentage has been recorded in sim 2 as per Fig 16. But from day 3 the company manages to decrease this delay in delivery time and from day 14 the delivery agent delivers on time to all customers. Here a downtrend is clearly noticeable which makes this model statically more significant.

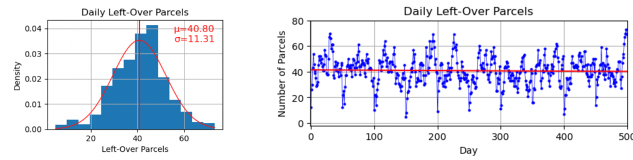

Fig. 15. Sim 2 Daily leftover parcels

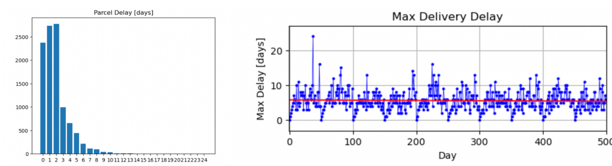

Fig. 16. Sim 2 Max delivery delay

3) Simulation 3(p-0.22, limit-30000, days-50, range-10): By taking the above parameters, the model generates parcels ranging from 1056 to 1132 over 50 days to 100 customers with a 10 simulation range.

Fig 17 explains the histogram and line plot of the daily working time for simulation 3. This project takes daily working time as a soft constraint and so here the average mean time is 180.30 with a significant downtrend.

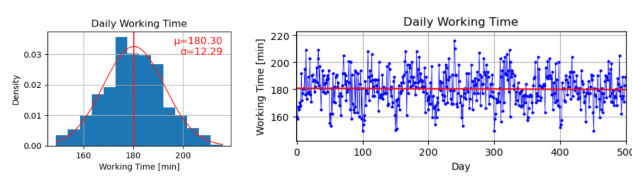

Fig. 17. Sim 3 Daily Working Time

The 30 km bike range has also been taken, which runs at its optimised condition as per Fig 18.

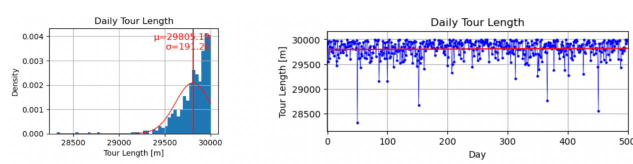

Fig. 18. Sim 3 Daily tour length

Compared to previous simulations here the over parcels have reached at the top, 45.30 with a significant downtrend, shown in Fig 19.

21.4% of parcel delay has been noticed on day 0 which slightly gets increased till day 3 as per Fig 20 but then a sudden downfall has been observed and till day 13 it has reached 0%.

4) Simulation 4(p-0.2, limit-35000, days-50, range-10): Now this study changes the bike limit to 35000 just to validate if the other p values work. For that, the study only takes only

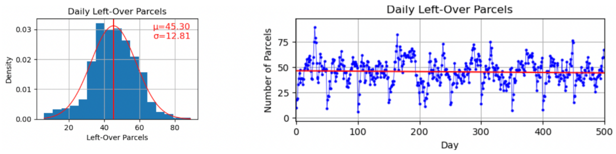

Fig. 19. Sim 3 Daily leftover parcels

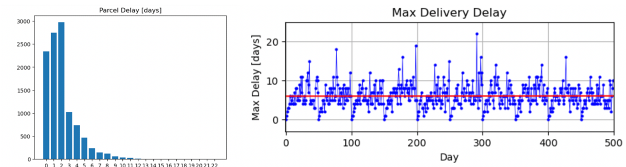

Fig. 20. Sim 3 Max delivery delay

p 0.2 and but at the beginning, only the mean daily working limit fails.

As per the given question, the delivery agent could work more than 180 minutes for some days but the average mean time must be in the range, shown in Fig 21.

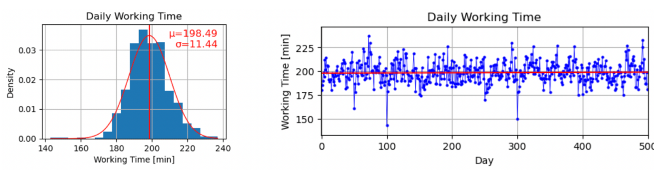

Fig. 21. Sim 4 Daily Working Time

The bike works at its optimum range as shown in Fig 22.

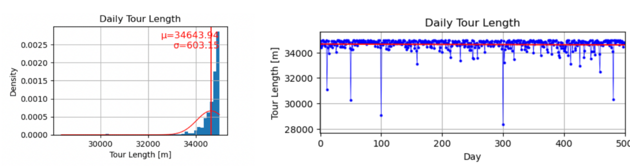

Fig. 22. Sim 4 Daily tour length

As per Fig 23, with the increase in the bike range, the daily leftover parcel significantly decreased to 25.42(average) with a statistically downtrend.

Connected with the downtrend the daily parcel delay has also noticeably dropped, although at day 0 it shows high but the company can able to manage it very quickly, shown in Fig 24.

IV. Result And Discussion

Simulation 4 could be the best-fit model out of the 4 simulations, but this study can not approve it as its business model for the "We-Doo" start-up as the mean daily working is beyond the limit and the company could not bear the cost of overtime. So comparing the first 3 simulations this study awards simulation 3 as the best simulation model for a good business value. Because here the average has slightly exceeded, 180.30 minutes which can be manageable and the

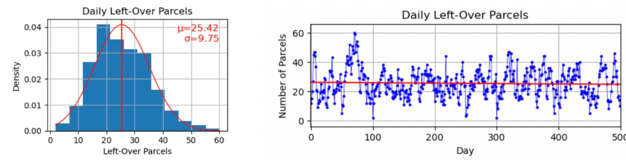

Fig. 23. Sim 4 Daily leftover parcels

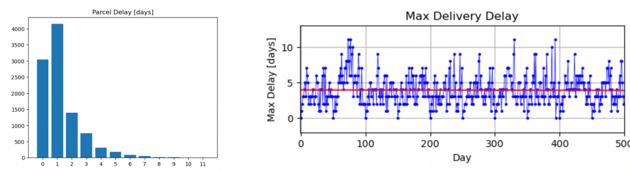

Fig. 24. Sim 4 Max delivery delay

crucial parameter(daily delivery delay) is better than the other two models as shown in Fig 25. Just like on day 0 simulation 3 starts with a 21.4% delivery delay but with the maximum number of total parcels. Although, on day 5 it is 4.2 when others have less in percentage on average this model wins throughout the time. In conclusion, the author would like to use another delivery agent so that the company could use a better range of bikes which would neutralise the daily left leftover parcels amount and daily parcel delay percentage as well. In that way they don't need to think about the over-time cost and customer satisfaction would also be there as well.

Delivery Delay	p=0.20 [9998 parcels] 30Km Bike Range	p=0.21 [10487 parcels] 30Km Bike Range	p=0.22 [10954 parcels] 30Km Bike Range	p=0.20 [9998 parcels] 35Km Bike Range
None	22.5%	22.6%	21.4%	30.5%
1 day	28.6%	26.0%	25.1%	41.5%
3 days	8.8%	9.5%	9.3%	7.6%
5 days	4.0%	4.2%	4.2%	1.8%
10 days	0.2%	0.3%	0.3%	0.1%
13 days	0.1%	0.1%	0.0%	0.0%
After 13 days	0.0%	0.0%	0.0%	0.0%

Fig. 25. Delivery Delay per Parcel depending on bike range (p=0.2)

Acknowledgment

This study would like to express all gratitude to Professor Dr Christian Horn for providing the simulation code and all the materials which have been used to accomplish this study.

References

[1] G. Perboli, M. Rosano and Q. Wei, "A Simulation-Optimization Approach for the Management of the On-Demand Parcel Delivery in Sharing Economy," in IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 8, pp. 10570-10582, Aug. 2022, doi: 10.1109/TITS.2021.3094851.

[2] P. Kowalik, G. Sobecki, P. Bawoł, and P. Muzolf, "A Flow-Based Formulation of the Travelling Salesman Problem with Penalties on Nodes," Sustainability, vol. 15, no. 5, p. 4330, Feb. 2023, doi: 10.3390/su15054330.

[3] Agryzkov, T., Oliver, J.L., Tortosa, L., Vicent, J.F. (2014). A Method to Triangulate a Set of Points in the Plane. In: Murgante, B., et al. Computational Science and Its Applications – ICCSA 2014. ICCSA 2014. Lecture Notes in Computer Science, vol 8580. Springer, Cham. https://doi.org/10.1007/978-3-319-09129-7_25